WPLoss:面向类别不平衡数据的加权成对损失

姚佳奇，徐正国，燕继坤，王科人

(盲信号处理重点实验室，成都 610041)

摘　要：类别不平衡数据是指不同类别的样本数目差异很大，AUC (Area Under the ROC Curve) 是衡量不平衡数据分类器性能的一个重要指标，由于 AUC 不可微，研究者提出了众多替代成对损失函数优化 AUC。成对损失的样本对数目为正负样本数目的乘积，大量成对损失较小的正负样本对影响了分类器的性能。针对这一问题，提出了一种加权的成对损失函数 WPLoss，通过赋予成对损失较大的正负样本对更高的损失权重，减少大量成对损失较小的正负样本对的影响，进而提升分类器的性能。在 20newsgroup 和 Reuters-21578 数据集上的实验结果验证了 WPLoss 的有效性，表明 WPLoss 能够提升面向不平衡数据的分类器的性能。

关键词：不平衡分类；成对损失；AUC 优化

中图分类号：TP391　　doi: 10.19734/j.issn.1001-3695.2020.02.0041

Wploss: weighted pairwise loss for class-imbalanced datasets

Yao Jiaqi, Xu Zhengguo, Yan Jikun, Wang Keren

(*National Key Laboratory of Science & Technology on Blind Signal Processing*, *Chengdu* 610041, *China*)

Abstract: Class-Imbalanced data refers to the large difference in the number of samples in different classes. AUC (area under the ROC curve) is an important metric to measure the performance of classifiers on the imbalanced datasets. Since AUC is not differentiable, researchers have proposed many surrogate pairwise loss functions to optimize AUC. The number of pairwise losses is the product of the number of positive and negative samples. Many positive and negative pairs with small pair loss affect the performance of classifiers. To solve this problem, this paper proposes a weighted pairwise loss function WPLoss. By assigning higher loss weights to the positive and negative samples with higher pairwise losses, WPLoss reduces the impact of positive and negative sample pairs with smaller pairwise losses. The experimental results on 20 newsgroup and Reuters-21578 datasets verify the validity of WPLoss, indicating that WPLoss can improve the performance of the classifier for class-imbalanced data.

Key words: class-imbalanced classification; pairwise loss; AUC optimization

0　引言

类别不平衡数据是指不同类别的样本数目差异较大，如图 1 所示，由 (x_1,x_2) 构成二维样本集，其中红色表示少数样本的类别，灰色表示多数样本的类别。类别样本数目的不平衡导致以优化 0-1 替代损失函数为目标的分类器失灵，因为分类器倾向于将所有样本预测为样本数目较多的类别[1]。而在实际应用中，通常样本数目较少的类别是用户感兴趣的类别，如在信用卡欺诈检测应用中，需要从大量账号中挑选出哪些账号可能存在欺诈行为，而有欺诈行为的账号属于少数，但是对银行却至关重要[2]；再如从海量文本数据中挑选出用户感兴趣或者认为重要的文本[3]；以及目标检测任务中前景和背景的分类[4]等。

类别不平衡分类的处理方法大体上可以分为两大类，分别是数据层面的处理方法和算法层面的处理方法，如图 2 所示。

数据层面的处理方法通过样本的重采样处理类别不平衡的问题，包括降采样算法、升采样算法和类别重组方法。降采样算法是指通过减少多数类别的样本实现类别平衡的算法，最简单的处理方法是随机降采样多数类别样本。降采样算法减少了训练样本的数目，从而提升了算法训练速度，但是损失了未未被采样到的多数类别样本的信息。针对降采样算法导致的样本信息损失的问题，Liu 等人提出了随机降采样多个多数类别的样本集，训练多个分类器然后集成[5]，另一些研究者研究了基于多数类别样本聚类的降采样算法[6, 7]。与降采样算法相反，升采样算法则通过增加少数类别的样本实现类别平衡的算法，最简单的处理方法是随机升采样少数类别的样本。随机升采样少数类别样本由于噪声样本的存在，导致分类器容易产生过拟合。Chawla 等人提出了通过邻近样本合成产生新的少数类别样本的 SMOTE 算法[8]。随着生成对抗网络(generative adversarial networks, GAN)的提出，研究者提出了一系列基于 GAN 生成少数类别样本的方法[9, 10]。Shicai Yang 提出了一种在场景分类任务中的类别重组方法(label shuffling)，首先按照最多数类别的样本数生成一个随机列表，其他类别则通过取其对应样本数的余数选取样本[11]。

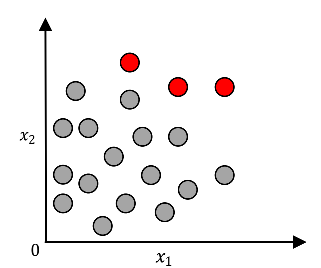


图 1　不平衡数据分布示意图

Fig. 1　Unbalanced data distribution diagram

收稿日期：2020-02-17；**修回日期：**2020-04-13

作者简介：姚佳奇(1992-)，男，山西侯马人，博士研究生，主要研究方向为自然语言处理，机器学习；徐正国(1985-)，男，湖北荆州人，工程师，博士，主要研究方向为网络协议分析，数据挖掘；燕继坤(1973-)，男，陕西周至人，高级工程师，博导，博士，主要研究方向为智能信息处理；王科人(1986-)，男，工程师，博士，主要研究方向为数据挖掘.

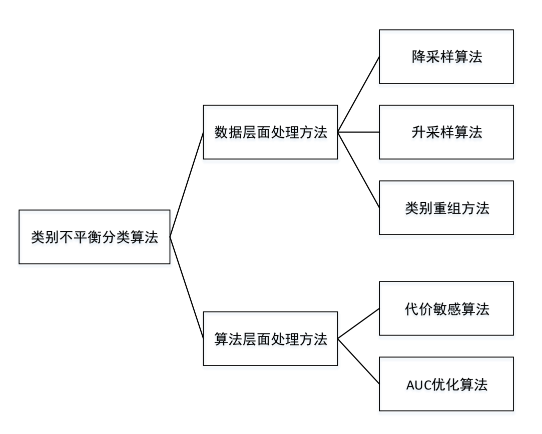


图 2　类别不平衡分类的处理方法分类

Fig. 2　Taxonomy of methods for class-imbalanced classification

算法层面处理方法则通过改变分类算法处理类别不平衡的问题，主要包括代价敏感类算法和 AUC 优化类算法。代价敏感类算法通过赋予不同类别不同的损失权重，提升分类器在处理不平衡数据的分类性能[12]，损失权重通常由不同类别的样本数目或者混淆矩阵确定。Lin 等人在代价敏感的基础上提出了 Focal Loss 损失函数，通过增加难以分类正确的样本损失权重进一步提升了分类器的性能[13]。AUC(area under curve)是衡量不平衡分类算法的一个关键指标，由于 AUC 不可直接优化，研究者提出了众多替代成对损失(surrogate pairwise loss)函数，包括指数损失(exponential loss)[14][15]，合页损失(hinge loss)[16][17]和最小二乘损失(least square loss)[18]等。

在计算 AUC 替代成对损失函数时，需要计算正负样本数乘积个成对损失，导致了大量成对损失较小的样本对占据了加大比重，从而主导了分类器梯度下降的方向。本文提出了一种加权成对损失 WPLoss(weighted pairwise loss)，通过赋予成对损失较大的样本对更大的损失权重，使分类器着重优化难以区分的正负样本对，进而提升分类器的性能。在公开数据集 20newsgroup 和 Reuters-21578 数据集上的实验结果，验证了本文提出方法的有效性，表明 WPLoss 提升了原始的 AUC 替代成对损失函数的分类性能，而且与其他不平衡分类算法相比，也具有较优的性能。

1　WPLoss:加权成对损失

本文提出的加权成对损失 WPLoss 是一种加权的 AUC 替代成对损失，旨在增加成对损失较大的正负样本对的损失权重，使得分类器着重优化难以区分的正负样本对。本节首先介绍了 AUC 优化方法，然后描述和分析了本文提出的 WPLoss。

1.1　AUC 优化

AUC 是 ROC(receiver operating characteristic)曲线下的面积。ROC 曲线的横坐标是假阳率(false positive rate, FPR)，纵坐标是真阳率(true positive rate, TPR)，如图 3 所示。

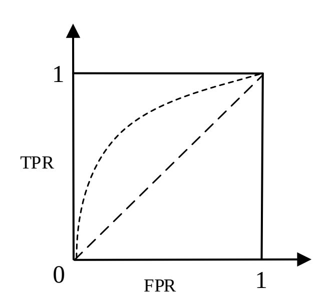


图 3　ROC 曲线示意图

Fig. 3　Roc curve

令 $X=\{x_1,x_2,\cdots,x_m,x_{m+1},\cdots,x_{m+n}\}$，其中前 m 个样本为正类样本，后 n 个样本为负样本，f 表示分类函数，tp 为为 f 正确预测为正类的样本个数，fp 为 f 错误预测为正类的样本个数，则 $FPR=\frac{fp}{n}$，$TPR=\frac{tp}{m}$。AUC 的形式定义如下所示。

$$AUC(f)=\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}I(f(x_i)>f(x_{m+j}))\tag{1}$$

其中 I 为示性函数，当括号内的表达式成立时值为 1，否则为 0。AUC 的取值范围在 $[0.5,1]$ 之间，值越大表明分类器的性能越好。从 AUC 的定义可以看出，分类器的 AUC 值是非凸和离散的，从而无法直接优化。

在实际应用中，研究者采用替代成对损失函数实现 AUC 的优化：

$$R(f)=\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\varphi(f(x_i)-f(x_{m+j}))\tag{2}$$

其中 φ 凸函数，如：

指数损失　　$\varphi(t)=e^{-t}$

逻辑斯蒂损失　　$\varphi(t)=\ln(1+e^{-t})$

合页损失　　$\varphi(t)=\max(0,1-t)$

最小二乘合页损失　　$\varphi(t)=\max(0,1-t)^2$

Gao 等人研究了替代成对损失与 AUC 优化的一致性问题，指出指数损失、逻辑斯蒂损失和最小二乘合页损失具有与 AUC 优化的一致性，而合页损失不具有[19]。

1.2　加权成对损失

从 AUC 的替代成对损失的定义式(2)可以看出，$R(f)$ 是正负样本数乘积 mn 个样本对距离的平均值。然而，不同的样本对有不同的距离，距离较小的样本对为较难区分的正负样本对，如图 4 中的样本对 1，而距离较大的样本对为较易区分的正负样本对，如图 4 中的样本对 2。式(2)中定义的 AUC 的替代成对损失采用了算术平均值，将导致大量的较容易区分的样本对占据损失的主要部分，从而主导了梯度方向。因而，本文提出了一种加权成对损失 WPLoss，使分类器着重优化难以区分的样本对。

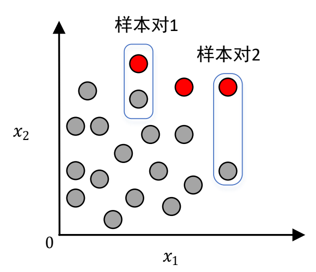


图 4　样本对距离示意图

Fig. 4　Pairwise distance diagram

令 $p_{ij}=\varphi(f(x_i)-f(x_{m+j}))$，则 WPLoss 的形式定义如下所示。

$$\alpha_{ij}=\frac{\exp(p_{ij})}{\sum_{i=1}^{m}\sum_{j=1}^{n}\exp(p_{ij})}\tag{3}$$

$$WPLoss(f)=\sum_{i=1}^{m}\sum_{j=1}^{n}\alpha_{ij}\cdot p_{ij}\tag{4}$$

WPLoss 采用 softmax 函数(式(3))加权不同样本对的成对损失，并实现归一化。WPLoss 给予较小距离的样本对较大的权重，较大距离的样本对较小的权重，使得优化算法着重优化难以区分的正负样本对，进而提升分类器的性能。同时，WPLoss 随着训练的进程，依据样本对的成对损失大小动态调整损失权重。

2　实验

针对不平衡数据的分类问题，本文提出了加权成对损失

WPLoss。本节通过两个公开的数据集上的实验，验证了 WPLoss 的有效性

2.1　数据集

a) 20newsgroup 数据集

20newsgroup(http://qwone.com/jason/20Newsgroups/) 包括大约 20,000 篇文档，大约被平均分成 20 组，即对应着 20 个类别，本文指定一个类别为正类，其余类别为负类，构造了 20 个二分类的数据集。

b) Reuters-21578 数据集

Reuters-21578 是一个公开的新闻数据集[20]，本文选择了样本数最多的十个类别，分别是 acq、crude、earn、grain、interest、money-fx、 money-supply、ship、sugar 和 trade，然后以某一类为正类，其余类别为负类，构造了 10 个二分类的数据集。

2.2　基础模型

本文采用卷积神经网络(convolutional neural network, CNN)作为文本特征提取器[22]，整体架构如图 5 所示，首先将文本使用词向量表示，然后依次经过卷积层、池化层和全连接层获取最终的特征向量。

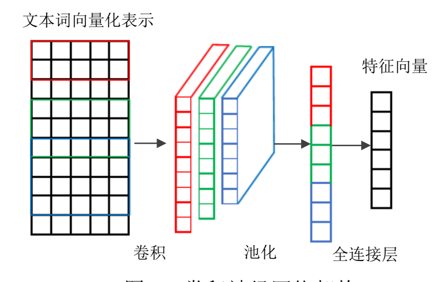


图 5　卷积神经网络架构

Fig. 5　Architecture of convolutional neural network

卷积网络的参数配置如表 1 所示。此外，本文采用了在谷歌新闻数据集上训练得到的词向量初始化卷积网络中的词向量矩阵[22]，并采用 Adam 优化算法[23]。

表 1　卷积神经网络的配置

Tab. 1　Settings of convolutional neural network

参数名称	参数设置
卷积核尺寸	3×300,4×300,5×300
卷积核个数	256
激励函数	Relu
池化函数	最大池化
随机失活率	0.5

2.3　对比方法

本文对比了原始 CNN，数据层面处理方法中的类别重组方法，算法层面处理方法中的代价敏感类方法，如偏置 CNN 和 Focal-Loss 方法，以及 AUC 优化方法。下面分别介绍每种方法的具体设定。

a) 原始 CNN

原始 CNN 方法忽略类别不平衡的问题，在使用 CNN 提取文本特征向量后，经过激活函数为 softmax 的全连接层，获取不同类别的概率，最后使用与真实标签交叉熵(Cross-Entropy)计算损失。令 $y_i \in \{0,1\}$ 第 i 个样本 x_i 的真实标签，p_i 表示分类器预测为正类的概率，则该样本的交叉熵损失为

$$CE(f(x_i)) = -y_i \log(p_i) - (1-y_i)\log(1-p_i) \tag{5}$$

b) 类别重组

类别重组在原始 CNN 的基础上，重采样了正类样本。首先按照负类的样本数生成一个随机列表，正类则使用该随机列表通过取正类样本数的余数采样。

c) 偏置 CNN

偏置(Biased)CNN 在原始 CNN 的基础上，令正类样本的损失权重为 1，负类样本的损失权重为正负类样本数目比 α，如下所示。

$$BiasedCE(f(x_i)) = -y_i \log(p_i) - \alpha(1-y_i)\log(1-p_i) \tag{6}$$

d) Focal-Loss

Focal-Loss 在偏置 CNN 的基础上，改造了交叉熵损失，如下所示。

$$FL(f(x_i)) = \\ -y_i(1-p_i)^{\gamma}\log(p_i) - \alpha(1-y_i)p_i^{\gamma}\log(1-p_i) \tag{7}$$

从 Focal-Loss 的定义可以看出，Focal-Loss 拓展了交叉熵损失的定义，当 $\gamma=0$ 时即为交叉熵损失，本文在实验时，取 $\gamma=2$ 。

e) AUCloss

AUCloss 为 AUC 替代成对损失的方法，在实验中，本文采用了指数损失，即令式(2)中的 φ 为 $\varphi(t)=e^{-t}$ 。

f) WPLoss

WPLoss 即本文提出的方法，为了验证 WPLoss 的有效性，采用与 AUCloss 的设定一样采用指数损失。

2.4　实验结果及分析

令 TP 表示正确预测为正类的样本个数，FP 表示错误预测为正类的样本个数，FN 为错误预测为负类的样本个数，则精确率 $P=\frac{TP}{TP+FP}$，召回率 $R=\frac{TP}{TP+FN}$，$F1=\frac{2PR}{P+R}$。本文采用了 F1 指标评估了不同算法的性能，20newsgroup 实验结果如表 2 所示，Reuters-21578 实验结果如表 3 所示，表中的不平衡率为负类样本数与正类样本数的比值，性能最优的结果使用粗体表示。

表 2　20newsgroup 实验结果

Tab. 2　Results on 20newsgroup

正类类别	不平衡率	原始 CNN	类别重组	偏置 CNN	Focal-Loss	AUCloss	WPLoss
alt.atheism	20.0481	0.8362	0.8580	0.8622	0.8625	0.8535	**0.8638**
comp.graphics	19.2269	0.8633	0.9142	0.9233	**0.9339**	0.9329	0.9336
comp.os.ms-windows.misc	18.5216	0.8979	0.9167	0.9279	**0.9414**	**0.9414**	0.9353
comp.sys.ibm.pc.hardware	19.6445	0.9169	0.9243	0.9519	0.9553	0.9553	**0.9569**
comp.sys.mac.hardware	19.5838	0.9210	0.9416	**0.9585**	0.9576	0.9496	0.9493
comp.windows.x	18.7698	0.9050	0.9416	0.9441	0.9469	0.9449	**0.9468**
misc.forsale	18.7141	0.8032	0.8784	**0.9322**	0.9072	0.9282	0.9262
rec.autos	18.8539	0.9489	0.9583	**0.9744**	0.9611	0.9641	0.9556
rec.motorcycles	18.5216	0.9746	0.9893	**0.9893**	0.9889	0.9839	0.9875
rec.sport.baseball	19.0530	0.9866	0.9983	0.9983	0.9981	0.9983	**1.0000**
rec.sport.hockey	18.8821	0.9949	0.9949	**0.9983**	0.9923	0.9933	0.9949
sci.crypt	18.7698	0.9554	0.9628	0.9573	0.9549	0.9429	**0.9666**
sci.electronics	18.3864	0.9396	0.9615	0.9597	0.9663	0.9613	**0.9632**

续表 2

正类类别	不平衡率	原始 CNN	类别重组	偏置 CNN	Focal-Loss	AUCloss	WPLoss
sci.med	18.9672	0.9672	0.9744	0.9694	0.9739	0.9779	**0.9698**
sci.space	19.4038	0.9572	0.9558	**0.9760**	0.9531	0.9741	0.9662
soc.religion.christian	18.5216	0.9982	0.9982	0.9982	0.9982	0.9982	**1.0000**
talk.politics.guns	18.6036	0.8905	0.8939	0.8862	0.8895	0.8885	**0.8923**
talk.politics.mideast	18.8539	0.9149	0.9287	**0.9276**	0.9293	0.9283	0.9267
talk.politics.misc	19.9223	0.6512	0.7375	0.7344	0.7540	0.7540	**0.7688**
talk.religion.misc	18.9672	0.6515	0.6877	0.7186	0.7357	0.7167	**0.7466**
平均值		0.8987	0.9208	0.9294	0.9300	0.9294	**0.9325**

表 3 Reuters-21578 实验结果

Tab. 3 Results on Reuters-21578

正类类别	不平衡率	原始 CNN	类别重组	偏置 CNN	Focal-loss	AUCloss	WPLoss
acq	3.0547	0.9711	0.9792	0.9730	0.9720	0.9698	**0.9854**
crude	18.1355	0.8889	0.8818	0.8876	0.8708	**0.9034**	0.8994
earn	1.2436	0.9885	0.9880	0.9849	0.9894	0.9921	**0.9940**
grain	17.3653	0.9462	0.9513	0.9270	0.9407	0.9457	**0.9621**
interest	35.8447	0.7174	0.7838	0.7376	0.7949	0.7826	**0.7945**
money-fx	13.5037	0.7379	0.7742	0.7572	0.7439	0.7365	**0.7799**
money-supply	49.7009	0.7636	0.8182	0.8000	0.7536	0.8000	**0.9000**
ship	48.8487	0.6444	0.7482	0.7237	0.7361	0.7114	**0.7500**
sugar	64.1868	0.8679	0.8966	0.8308	0.8182	0.8571	**0.8966**
trade	19.1085	0.8756	0.8889	0.8308	0.8475	0.8815	**0.8950**
平均值		0.8402	0.8710	0.8368	0.8467	0.8580	**0.8857**

从实验结果中可以看出原始 CNN 在两个数据集上的性能都较差，尤其是当类别极不平衡时，如在 Reuters-21578 数据集上，当正类类别为 money-supply 和 ship 等，说明以 0-1 损失为优化目标的算法在面对不平衡数据分类时性能受限。类别重组算法在两个数据集上的平均性能都较优，但是由于升采样了正类样本，相当于增加了训练样本数，从而导致相比其他算法所需的训练时间更长。而代价敏感类的算法 Focal-Loss 相对于偏置 CNN 算法的性能有所提升，表明 Focal-Loss 对偏置 CNN 改进的有效性。

实验结果同时表明，本文提出的 WPLoss 在绝大部分数据集上的性能优于其他方法，并且在两个数据集上的平均性能取得最优结果，从而验证了 WPLoss 的有效性。与原始的 AUCLoss 相比，WPLoss 几乎在所有数据集上都取得了较优的性能，表明 WPLoss 通过着重优化难以区分的正负样本对提升了分类器的性能。当正负类别极为不平衡时，WPLoss 与 AUCLoss 相比性能优势更加明显。如 Reuters-21578 中，正类类别为 money-supply 时，不平衡率为 49.7009，WPLoss 与 AUCLoss 相比提升了 12.8%；正类类别为 ship 时，不平衡率为 48.8487，WPLoss 与 AUCLoss 相比提升了 5.4%；正类类别为 sugar 时，不平衡率为 64.1868，WPLoss 与 AUCLoss 相比提升了 4.6%。

3 结束语

本文提出了一种面向不平衡数据的加权成对损失 WPLoss，通过赋予距离较小的正负样本对更大的损失权重，使得优化算法着重优化难以区分的正负样本对，从而提升分类器的性能。在公开数据集 20newsgroup 和 Reuters-21578 上的实验结果表明，WPLoss 不仅仅相对原始的成对损失有较优的性能，同时，与其他处理不平衡数据的方法相比，也取得了较优的性能，尤其在不平衡率较高的时候。

参考文献：

[1] 向鸿鑫，杨云. 不平衡数据挖掘方法综述 [J]. 计算机工程与应用, 2019, 55 (04): 6-21. (Xiang Hongxin, Yang Yun. Survey on Imbalanced Data Mining Methods [J]. Computer Engineering and Applications, 2019, 55 (04): 6-21.)

[2] García V, Marqués A I, Sánchez J S. Exploring the synergetic effects of sample types on the performance of ensembles for credit risk and corporate bankruptcy prediction [J]. Information Fusion, 2019, 47: 88-101.

[3] Padurariu C, Breaban M E. Dealing with Data Imbalance in Text Classification [J]. Procedia Computer Science, 2019, 159: 736-745.

[4] Oksuz K, Cam B C, Kalkan S, *et al.* Imbalance Problems in Object Detection: A Review [J]. arXiv: Computer Vision and Pattern Recognition, 2019.

[5] Liu, Xuying, Wu Jianxin, Zhou Zhihua. Exploratory Undersampling for Class-Imbalance Learning [J]. IEEE Trans on Systems Man & Cybernetics Part B, 2009, 39 (2): 539-550.

[6] Lin Weichao, Tsai C, Hu Yahan, *et al.* Clustering-based undersampling in class-imbalanced data [J]. Information Sciences, 2017, 409/410: 17-26.

[7] Ofek N, Rokach L, Stern R, *et al.* Fast-CBUS: a fast clustering-based undersampling method for addressing the class imbalance problem [J]. Neurocomputing, 2017, 243: 88-102.

[8] Chawla N V, Bowyer K W, Hall L O, *et al.* SMOTE: synthetic minority over-sampling technique [J]. Journal of Artificial Intelligence Research, 2002, 16 (1): 321-357.

[9] Fiore U, De Santis A, Perla F, *et al.* Using generative adversarial networks for improving classification effectiveness in credit card fraud detection [J]. Information Sciences, 2019, 479: 448-455.

[10] Douzas G, Bacao F. Effective data generation for imbalanced learning using conditional generative adversarial networks [J]. Expert Systems with applications, 2018, 91: 464-471.

[11] Shicai Yang. Several tips and tricks for ImageNet CNN training. Technique Report pages 1-12, 2016.

[12] Zhou Z H. Cost-sensitive learning [M]. Modeling decision for artificial

intelligence. Berlin, Heidelberg: Springer, 2011: 17-18.

[13] Lin, Tsung-Yi, Goyal, Priya, Girshick, Ross, *et al.* Focal Loss for Dense Object Detection [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, PP (99): 2999-3007.

[14] Freund Y, Iyer R D, Schapire R E, *et al.* An efficient boosting algorithm for combining preferences [J]. Journal of Machine Learning Research, 2003, 4 (6): 933-969.

[15] Rudin C, Schapire R E. Margin-based Ranking and an Equivalence between AdaBoost and RankBoost [J]. Journal of Machine Learning Research, 2009: 2193-2232.

[16] Ulf B, Tobias S. AUC maximizing support vector learning. [C]// Proc of ICML, 2005.

[17] Joachims T. A support vector method for multivariate performance measures [C]// Proc of ICML, 2005: 377-384.

[18] Wei Gao, Rong Jin, Shenghuo Zhu, *et al.* One-Pass AUC Optimization [J]. arXiv: Learning, 2013.

[19] Gao Wei, Zhou Zhihua. On the consistency of AUC pairwise optimization [C]// Proc of 24th International Joint Conference on Artificial Intelligence. 2015, 939-945.

[20] Yang Yiming and Liu Xin. A re-examination of text categorization methods [C]// Proc of 22nd Annual International SIGIR Conference on Research and Development in Information Retrieval, 1999, 42-49

[21] Kim Y. Convolutional neural networks for sentence classification [J]. arXiv preprint arXiv: 1408. 5882, 2014.

[22] Mikolov T, Sutskever I, Chen K, *et al.* Distributed representations of words and phrases and their compositionality [C]// Proc of NIPS, 2013, 3111-3119.

[23] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization [J]. arXiv: Learning, 2014.